REMARKS

Claims previously in issue:	1-39
Claims amended:	1, 9, 14, 22, 27, 35
Claims cancelled:	None
Claims added:	None
Claims rejected	1-39
Claims remaining in issue:	1-39

The Examiner has again rejected the present claims as anticipated under 35 USC 102(a) in light of the reference to Hoffberg *et al.* (U.S. Patent No. 6,400,996 B1). Applicants respectfully traverse this rejection.

Applicants again emphasize that the reference is not applicable art, as would be recognized by anyone of ordinary skill in the art. The Examiner persists in confusing the use of the depiction of a graph (meaning a "*Diagram showing varying quantities: a diagram used to indicate relationships between two or more variable quantities. The quantities are measured along two axes, usually at right angles. A graph may consist, for example, of a line joining points plotted between coordinates, a series of parallel bars or boxes, or a circle divided into wedges.*" See 2001 Encarta® World English Dictionary from Microsoft Corporation) in Hoffberg (see, *e.g.*, Fig. 3) with a data flow graph as defined by Applicants and used in the claims:

> Complex business systems typically process data in multiple stages, with the results produced by one stage being fed into the next stage. The overall flow of information through such systems may be described in terms of a directed data flow <u>graph, with vertices in the graph representing components (either data files or processes), and the links or "edges" in the graph indicating flows of data between components</u>.
> [p. 1, ll. 6-10; emphasis added]

As noted by Applicants previously, it is ancient and settled law that an applicant may be his or her own lexicographer. See MPEP 2111.01. Having defined the term "graph" in the specification, the Examiner cannot force a different meaning on the word.

In the present claims and application, the term "graph" is a noun defined in the specification (which meaning is also well known in the data processing arts), and would not be confused by one of ordinary skill in the art with a 2-dimensional relationship graph as depicted in Hoffberg, or with the alternative dictionary meaning of graph ("written symbol: a symbol, letter, or combination of letters used in writing to represent the smallest discrete unit of speech", 2001 Encarta® World English Dictionary from Microsoft Corporation), or with the Graf Zeppelin. To argue otherwise would be error.

Hoffberg mentions "graph" *per se* only once, when discussing a financial graph. Hoffberg mentions "graphical" (as in "graphical user interface"), "graphics" (as in images or icons), "demographics", "biography", "hypergraph", "bibliography", "Siggraph", "holography", "Eurographics", "cryptographic", "photographic", "telegraph", "electromygraph", "ultrasonography", "paragraph", and variants of these terms – but nothing about data flow graphs, either in words or in concept.

Hoffberg also depicts several 2-dimensional charts of data (*e.g.*, Fig. 3 of the reference, cited by the Examiner). Hoffberg's only description of Fig. 3 states: "FIG. 3 graphically shows the differences in seconds between total time for the prior art for each user". In context (see Figs. 1-14; there is no textual discussion of these figures, other than under the BRIEF DESCRIPTION OF THE DRAWINGS section), Hoffberg is presenting statistical data on the number of key presses saved (and thus of time saved) by various users using his new user interface. (As noted previously by Applicants, Hoffberg relates to an adaptive pattern recognition based control system and method; in essence, Hoffberg teaches a method for predicting what instructions or choices a user most likely will use based on previous inputs from the user, for example, for use in programming a VCR).

Thus, Hoffberg teaches nothing about data flow graphs having vertices representing components and links between components indicating flows of data between such components.

To make this difference clear to even casual readers, Applicants have amended the preamble of the independent claims to include explicitly the already implicit definition of "graph" in accordance with the specification.

Hoffberg also fails to teach anything about parameters, runtime or not, or about retrieving runtime parameters (the word "parameter" does not appear in Hoffberg at all; the Examiner

equates performance benchmarks *for* Hoffberg's user interface as being *parameters of* Hoffberg's invention – which is an incorrect characterization). The present invention formalizes the parameterization of data flow graphs to allow runtime parameters. Runtime parameters allow an application builder to defer the value of a parameter setting (*e.g.*, the key parameter of a sort function, file names, record formats, transform functions, *etc.*) to runtime (*i.e.*, the time an application is executed on a computer system); see p. 4, ll. 3-6.

Moreover, Hoffberg teaches nothing about determining whether the value for a runtime parameter is to be provided by user input. As the application makes clear, "the values of runtime parameters may be supplied by the end user or be derived from a combination of other runtime parameters or objects stored in an object repository" (p. 4, ll. 6-7). Thus, the implementing system needs to determine whether or not the value for a runtime parameter is to be provided by user input. Hoffberg does not discuss or teach this concept.

Furthermore, the "graphs" depicted by Hoffberg cannot be executed; they are simply static graphical presentations of statistical data on the number of key presses saved (and thus of time saved) by various users using Hoffberg's new user interface.

The following table summarizes the complete lack of teaching by Hoffberg of the elements of exemplary claim 1, as amended:

Exemplary Claim	**Hoffberg**
1. A method for executing a graph having <u>vertices representing components and links between components indicating flows of data between such components, the graph having</u> components with parameters, including:	• No teachings whatsoever re graphs having vertices representing components and links between components indicating flows of data between such components • No teachings re executing such graphs
(a) retrieving a runtime parameter for the graph at runtime execution of the graph, the runtime parameter having a value defined as determinable at runtime execution of the graph;	• No teachings re retrieving runtime parameters at any time • No teachings re runtime parameters having a value defined as determinable at runtime execution of the graph

(b) determining whether the value for the runtime parameter is to be provided by user input;	• No teachings re determining whether the value for the runtime parameter is to be provided by user input
(c) displaying a prompt to a user for receiving user input for every runtime parameter so determined;	• While Hoffberg may display a prompt for user input, it does not do so with respect to runtime parameters determined to require user input
(d) determining a first final parameter value based on user response to such prompt; and	• No teachings re determining a final first parameter value (which is taught by Applicants in one embodiment as "a determination … of the final value for the parameter, taking into account transformations of the input and dependencies and conditions based on other parameters"; p. 17, ll. 16-19)
(e) executing the graph using the first final parameter value as the value for the runtime parameter.	• No teachings re *executing* graphs at all.

Thus, Hoffberg utterly fails as a relevant reference, let alone an anticipatory reference, with respect to the invention as claimed. Accordingly, Applicants submit that all of the claims are allowable.